Project Overview

This project consists of two files:
1. **kwork.ipynb: Step 1** – Preparation, creation, and indexing of document embeddings using Sentence Transformers and FAISS for document search.
2. **app.py: Step 2** – A Streamlit-based interactive QA system that allows users to upload PDF files, ask questions, and receive answers using FAISS for document search and Cohere's API for text generation.

Step-by-Step Guide

Step 1: kwork.ipynb – Embedding and Indexing Documents

1. **Import Required Libraries:** This step includes importing essential libraries like Sentence Transformers (for creating embeddings), FAISS (for efficient search), and others necessary for file handling.

2. **Document Preparation:**
   - You will load documents or text data that need to be indexed. The text is split into paragraphs or sections for individual processing.

3. **Create Embeddings:**
   - Sentence Transformers model (likely all-MiniLM-L6-v2) is used to generate embeddings (vector representation) for each paragraph or section of the document.

4. **FAISS Index Creation:**
   - The embeddings are indexed using FAISS to allow for efficient similarity searches. This step saves the FAISS index for later retrieval in the app.

Step 2: app.py – Interactive QA System

1. Set Up Environment: Ensure you have the required libraries installed. Use the following pip command:

```
  bash
  pip install sentence-transformers faiss-cpu pymupdf cohere streamlit
```

2. Run the Application:
   - You can run this file using streamlit run app.py.

3. Upload PDF:
   - The application allows you to upload a PDF document. Upon uploading, the PDF is processed and its text extracted using pymupdf.

4. Embedding and Indexing:

- The text is split into sections or paragraphs, and embeddings are created using Sentence Transformers. These embeddings are indexed using FAISS for fast retrieval.

5. Ask a Question:

- Once the document is indexed, users can input a question. The system retrieves the most relevant sections from the document based on the query using FAISS search.

6. Generate Answer:

- After retrieving relevant sections, the system uses Cohere's API to generate an answer based on the context provided by the retrieved sections.

How to Use:

1. Run Step 1: Open the kwork.ipynb notebook and run the cells sequentially to generate and store document embeddings using Sentence Transformers. This will create a FAISS index.
2. Run Step 2: Start the Streamlit app by running app.py. You can upload a PDF, ask questions, and get responses generated using the system.